REMARKS

Applicant has carefully reviewed and considered the Office Action mailed on January 15, 2003, and the references cited therewith.

Claims 1, 30 and 31 are amended. Claims 32 and 33 have been cancelled. Claims 1-5, 8, 11-20, 23-31, 33-38, 41-46, 49, 52, 55-64, 67-77 are now pending in this application.

§112 Rejection of the Claims

Claims 32 and 33 were rejected under 35 USC § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicant regards as the invention. Claims 32 and 33 have been cancelled.

§102 Rejection of Claims 30-32, 34 and 35 / §103 Rejection of the Claims 33 and 36

Claims 30-32 and 34-35 were rejected under 35 USC § 102(b) as being anticipated by Gusefski et al. (U.S. Patent No. 5,202,972). Claim 32 has been cancelled and claims 30 and 31 amended. Claims 30, 31, 34 and 35, as amended, do distinguish over Gusefski et al. (U.S. Patent No. 5,202,972).

Claims 33 and 36 were rejected under 35 USC § 103(a) as being unpatentable over Gusefski et al. in view of Lange et al. Claim 33 has been cancelled. Claims 30, 31, 34, 35 and 36, as amended, do distinguish over Gusefski et al. in view of Lange et al.

Gusefski describes a system having processors and common master storage unit. Each processor includes cache memory. The common master storage unit includes master storage (102, 106), a second level of cache (27) and a storage controller (26) that arbitrates for access to master storage. Col. 4, lines 41-66. If a processor tries to retrieve data that it is not within its L1 cache, it accesses storage controller 26. If the master storage resource is being used by another processor, it waits for its turn. Eventually, it does access storage controller 26 and can retrieve data. The system described by Gusefski requires that each processor monitor for fetch conflicts in order to maintain cache coherency. Fig. 6 and col. 14, lines 23-58.

In contrast, Applicant teaches that it is possible to place a first identifier and a second identifier in his resource controller. By doing so, now each processor can determine if resource memory they are interested in has been accessed, and potentially modified, by another processor.

If so, data stored in the processor's cache is invalidated. Subsequent fetches are forced to, therefore, to go to resource memory. This is much simpler that monitoring for and reacting to fetch conflicts.

As the Examiner notes, Lange discusses the clearing of cache each time that a processor arbitrates for access to main memory store. This is a fairly simple mechanism. It has the disadvantage in that, if a processor wins consecutive arbitrations for main memory, its cache is still invalidated. Applicant avoids that problem by monitoring the present and previous owners of memory or sections of memory in main memory, and clearing cache in the processor accessing that memory only if it is not the last processor to access that memory or section of memory. Applicant respectfully submits that claims 30, 31, 34, 35 and 36, as amended, do distinguish over Gusefski et al. and Lange et al., either alone or together.

§103 Rejection of the Claims

Claims 1-5 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. (U.S. Patent No. 3,845,474) in view of Averill (U.S. Patent No. 5,313,591).

As discussed above, and as the Examiner notes, Lange discusses the clearing of cache each time that a processor arbitrates for access to main memory store. This is a fairly simple mechanism. It has the disadvantage in that, if a processor wins consecutive arbitrations for main memory, its cache is still invalidated. Applicant avoids that problem by monitoring the present and previous owners of memory or sections of memory in main memory, and clearing cache in the processor accessing that memory only if it is not the last processor to access that memory or section of memory.

Averill addresses the problem of designing computer busses to accommodate transactions which may take a variable amount of time to complete. Averill states that a cache coherency mechanism which relies on the flushing or purging of cache memory of other processors as part of a transaction originated by one of the processors is an example of such a transaction.

Claim 1 has been amended. Neither Lange nor Averill teach a system which discriminates between consecutive and disparate data owners in determining whether to reset a portion of memory associated with the processor, as is claimed in claims 1-5. Applicant

respectfully submits that claims 1-5, as amended, do distinguish over Lange et al. and Averill, either alone or together.

Claims 8, 11, and 14-19 were rejected under 35 USC § 103(a) as being unpatentable over Lange in view of Averill in further view of Gusefski et al.

As noted above, neither Lange nor Averill teach a system which discriminates between consecutive and disparate data owners in determining whether to reset a portion of memory associated with the processor, as is claimed in claims 8, 11, and 14-19.

Also as noted above, Gusefski describes a system having processors and common master storage unit. Each processor includes cache memory. The common master storage unit includes master storage (102, 106), a second level of cache (27) and a storage controller (26) that arbitrates for access to master storage. Col. 4, lines 41-66. If a processor tries to retrieve data that it is not within its L1 cache, it accesses storage controller 26. If the master storage resource is being used by another processor, it waits for its turn. Eventually, it does access storage controller 26 and can retrieve data. The system described by Gusefski requires that each processor monitor for fetch conflicts in order to maintain cache coherency. Fig. 6 and col. 14, lines 23-58.

Gusefski, therefore, also fails to monitor which processor last had access to the resource in order to determine whether to flush cache.

Applicant respectfully submits that claims 8, 11, and 14-19 do distinguish over Gusefski, Lange et al. and Averill, either alone or together.

Claim 12 was rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill and in further view Gusefski et al. and Tanenbaum (Structured Computer Organization). Claim 12 is dependent on claim 8 and inherits all the limitations of claim 8. Applicant respectfully submits that, based on the limitations in claim 8, claim 12 does distinguish over Gusefski, Tanenbaum, Lange and Averill, either alone or together.

Claim 13 was rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill and in further view Gusefski et al., Tanenbaum, and Georgiou et al. (U.S. Patent No. 4,633,394). Claim 13 is dependent on claim 8 and inherits all the limitations of claim 8. Applicant respectfully submits that, based on the limitations in claim 8, claim 13 does distinguish over Gusefski, Tanenbaum, Georgiou, Lange and Averill, either alone or together.

Claim 20 and 26-29 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view of Gusefski et al..

As noted above, neither Lange nor Averill teach a system which discriminates between consecutive and disparate data owners in determining whether to reset a portion of memory associated with the processor, as is claimed in claims 20 and 26-29.

Also as noted above, Gusefski describes a system having processors and common master storage unit. Each processor includes cache memory. The common master storage unit includes master storage (102, 106), a second level of cache (27) and a storage controller (26) that arbitrates for access to master storage. Col. 4, lines 41-66. If a processor tries to retrieve data that it is not within its L1 cache, it accesses storage controller 26. If the master storage resource is being used by another processor, it waits for its turn. Eventually, it does access storage controller 26 and can retrieve data. The system described by Gusefski requires that each processor monitor for fetch conflicts in order to maintain cache coherency. Fig. 6 and col. 14, lines 23-58.

Gusefski, therefore, also fails to monitor which processor last had access to the resource in order to determine whether to flush cache.

Applicant respectfully submits that claims 20 and 26-29 do distinguish over Gusefski, Lange et al. and Averill, either alone or together.

Claims 23-25 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view of Gusefski et al. and Tanenbaum. Claims 23-25 are dependent on claim 20 and inherit all the limitations of claim 20. Applicant respectfully submits that, based on the limitations in claim 20, claims 23-25 do distinguish over Gusefski, Tanenbaum, Lange and Averill, either alone or together.

Claims 37-38, 41 and 42 were rejected under 35 USC § 103(a) as being unpatentable over Gusefski et al. in view of Lange et al.

Gusefski and Lange are discussed above.

In contrast, Applicant teaches that it is possible to place a first identifier and a second identifier in his resource controller. By doing so, now each processor can determine if resource memory they are interested in has been accessed, and potentially modified, by another processor. If so, data stored in the processor's cache is invalidated. Subsequent fetches are forced to,

therefore, to go to resource memory. This is much simpler that monitoring for and reacting to fetch conflicts.

As the Examiner notes, Lange discusses the clearing of cache each time that a processor arbitrates for access to main memory store. This is a fairly simple mechanism. It has the disadvantage in that, if a processor wins consecutive arbitrations for main memory, its cache is still invalidated. Applicant avoids that problem by monitoring the present and previous owners of memory or sections of memory in main memory, and clearing cache in the processor accessing that memory only if it is not the last processor to access that memory or section of memory. Applicant respectfully submits that claims 37-38, 41 and 42 do distinguish over Gusefski et al. and Lange et al., either alone or together.

Claims 43-46 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view of Gusefski et al. Applicant respectfully submits that claims 43-46 do distinguish over Gusefski, Lange et al. and Averill, either alone or together, for the reasons given in the discussion of claim 8 above.

Claim 49 was rejected under 35 USC § 103(a) as being unpatentable over Gusefski et al. in view of Averill.

Averill and Gusefski are discussed above. As noted above, Averill does not teach a system which discriminates between consecutive and disparate data owners in determining whether to reset a portion of memory associated with the processor, as is claimed in claims 8, 11, and 14-19. Gusefski also fails to monitor which processor last had access to the resource in order to determine whether to flush cache. Claim 49 includes this limitation. Applicant respectfully submits that claim 49 does distinguish over Gusefski and Averill, either alone or together.

Claims 52, 55, 58-59, and 61-63 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill.

As noted above, neither Lange nor Averill teach a system which discriminates between consecutive and disparate data owners in determining whether to reset a portion of memory associated with the processor, as is claimed in claims 52, 55, 58-59, and 61-63. Applicant respectfully submits that claims 52, 55, 58-59, and 61-63 do distinguish over Lange et al. and Averill, either alone or together.

Claim 56 was rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view of Tanenbaum et al. Claim 56 is dependent on claim 52 and inherits all the limitations of claim 52. Applicant respectfully submits that, based on the limitations in claim 52, claim 56 does distinguish over Tanenbaum, Lange and Averill, either alone or together.

Claim 57 was rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view Tanenbaum et al, and Georgiou et al. Claim 57 is dependent on claim 56 and inherits all the limitations of claims 52 and 56. Applicant respectfully submits that, based on the limitations in claim 52 and 56, claim 57 does distinguish over Tanenbaum, Georgiou, Lange and Averill, either alone or together.

Claim 60 was rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view Gusefski et al. Claim 60 is dependent on claim 52 and inherits all the limitations of claim 52. Applicant respectfully submits that, based on the limitations in claim 52, claim 60 does distinguish over Gusefski, Lange and Averill, either alone or together.

Claims 64 and 70-73 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view of Gusefski et al. Applicant respectfully submits that claims 64 and 70-73 do distinguish over Gusefski, Lange et al. and Averill, either alone or together, for the reasons given in the discussion of claim 8 above.

Claims 67-69 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill, and in further view of Gusefski et al. and Tanenbaum et al. Claims 67-69 are dependent on claim 64 and inherit all the limitations of claim 64. Applicant respectfully submits that, based on the limitations in claim 64, claims 67-69 do distinguish over Gusefski, Tanenbaum, Lange and Averill, either alone or together.

Claims 74-77 were rejected under 35 USC § 103(a) as being unpatentable over Lange et al. in view of Averill.

As noted above, neither Lange nor Averill teach a system which discriminates between consecutive and disparate data owners in determining whether to reset a portion of memory associated with the processor, as is claimed in claims 74-77. Applicant respectfully submits that claims 74-77 do distinguish over Lange et al. and Averill, either alone or together.

CONCLUSION

Applicant respectfully submits that the claims are in condition for allowance. Reconsideration of all pending claims and notification of allowance is earnestly requested. The Examiner is invited to telephone Applicant's attorney at (612) 373-6909 to facilitate prosecution of this application.

If necessary, please charge any additional fees or credit overpayment to Deposit Account No. 19-0743.

Respectfully submitted,

MARK D. RUSTAD

By his Representatives,

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
P.O. Box 2938
Minneapolis, MN 55402
(612) 373-6909

Date July 15, 2003

By Thomas F. Brennan
Thomas F. Brennan
Reg. No. 35,075

CERTIFICATE UNDER 37 CFR 1.8: The undersigned hereby certifies that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail, in an envelope addressed to: Commissioner of Patents, Washington, D.C. 20231, on this 15th day of July, 2003.

Candis B. Buending
Name

Candis Buending
Signature